

**Team :** Bad-Apples

**Features to implement in next sprint:**

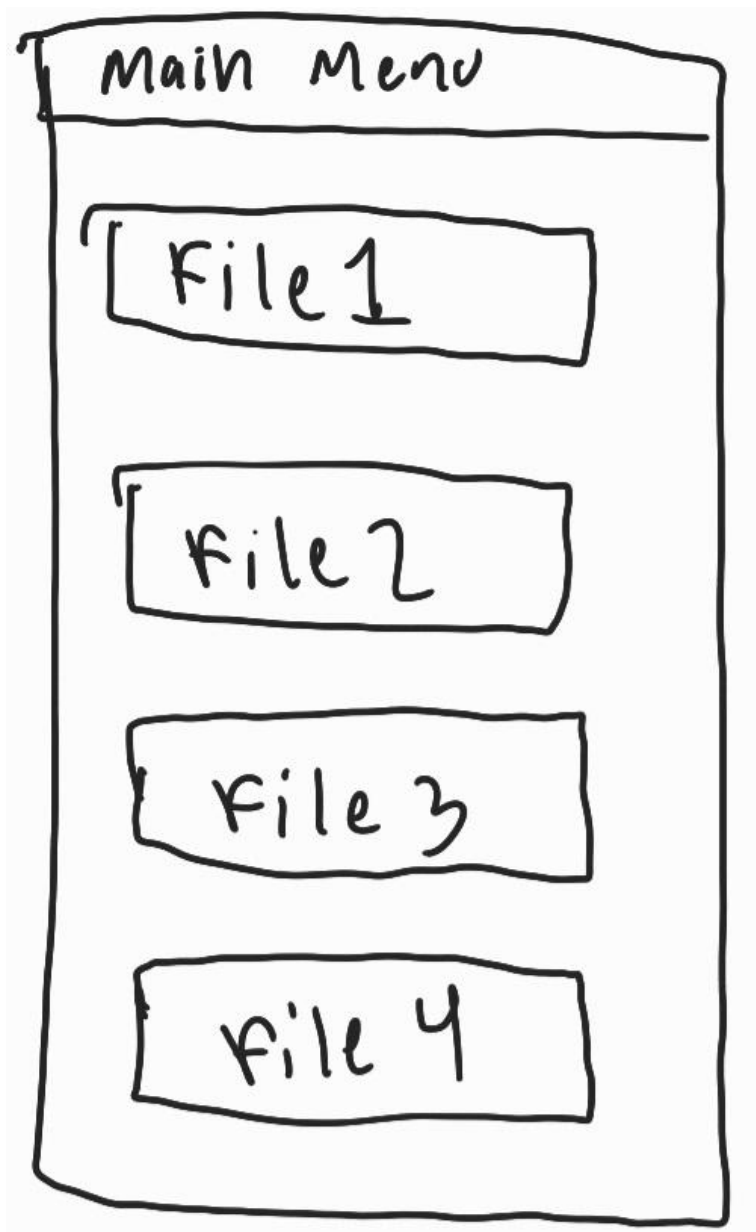
- 1) Update, Insert, Delete, backup/import relating to Date.
- 2) Update, Insert, Delete, backup/import relating to Boroughs.
- 3) Update, Insert, Delete, backup/import relating to Time.
- 4) Update, Insert, Delete, backup/import relating to Vehicle Class.
- 5) Update, Insert, Delete, backup/import relating to Active vehicles.
- 6) Update, Insert, Delete, backup/import relating to Streets.

**User Stories:**

- 1) As a user, I want to Update, Insert, Delete rows relating to date.
- 2) As a user, I want to Update, Insert, Delete rows relating to Boroughs.
- 3) As a user, I want to Update, Insert, Delete rows relating to Time.
- 4) As a user, I want to Update, Insert, Delete rows relating to Vehicle Class.
- 5) As a user, I want to Update, Insert, Delete rows relating to Active vehicles.
- 6) As a user, I want to Update, Insert, Delete rows relating to Streets.
- 7) As a user I want to back up and import version(s) of data previously stated.

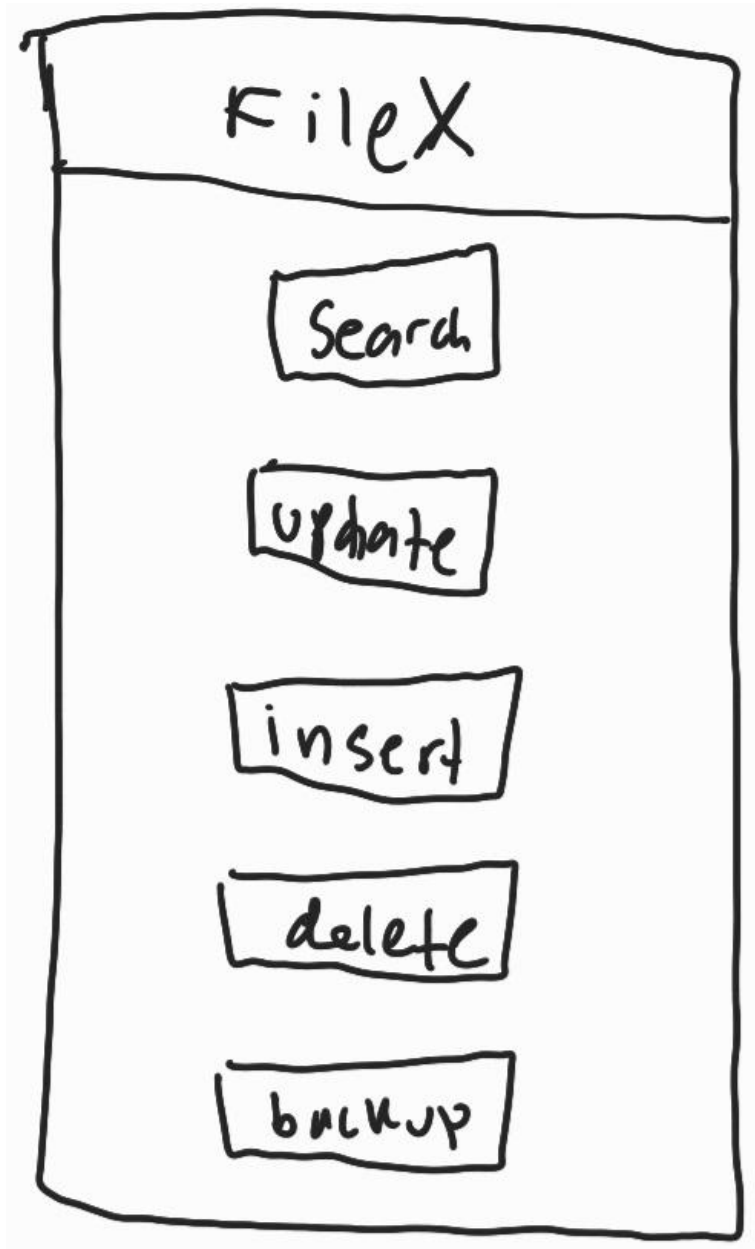
**Main Activity:**

- All 6 data sets show up in options



**File Activity:**

- Display all actions to be done on the file
- Have the ability to view the file
- Choose to back up the file and then import the file



**Search Activity:**

- Display result from search on fileX

A hand-drawn sketch of a mobile application interface. It consists of a rounded rectangle with a header bar at the top containing the word "Search". Below the header is a horizontal line. The main content area contains the text "FileX", followed by "H of Rides on" and "date:" on the next line. At the bottom of the main content area is the word "Result".

### Update Activity:

- Choose the row
- Let the user write the update
- Update button to update

A hand-drawn sketch of a mobile application screen. At the top, there is a header bar with the word "Update" written in a casual, handwritten font. Below the header, the text "file x" is centered. Underneath "file x", there are two input fields. The first is labeled "choose row:" followed by a horizontal line representing a text input. The second is labeled "update:" followed by a horizontal line representing a text input. At the bottom of the screen, there is a rectangular button with the word "update" written inside it in a handwritten style.

**Insert Activity:**

- Insert all 3 columns for the row
- Insert Button to insert

The diagram illustrates a hand-drawn interface for an 'Insert' activity. It consists of a large rectangular container with a rounded top. At the very top, within a small header area, is the word 'Insert'. Below this, the text 'File X' is centered. Underneath 'File X' is the text 'Sample Row'. Following 'Sample Row' are three lines, each starting with a column label followed by a horizontal line: 'column 1', 'column 2', and 'column 3'. Below these labels is a vertical ellipsis (three dots). At the bottom of the container is a rectangular button with the word 'Insert' written inside it.

**Delete Activity:**

- Choose what row to delete
- Delete button to delete

A hand-drawn sketch of a form for a delete activity. The form is a vertical rectangle with a header section at the top. The header section contains the word "delete" in a handwritten font. Below the header, the text "file x" is written. Underneath that, the text "choose Row: \_" is written, indicating a selection point. At the bottom of the form, there is a rectangular button with the word "delete" written inside it.

## Test Cases:

- **Feature 1 Test Cases** : as a user, I want to Update, Insert, Delete rows relating to Date.
  - **Test case 1** : as a user, in the update page, I input what row I want to edit. Then I enter the desired update in the update text box. Then I push the update button.  
Correct Output : The app displays a toast message saying update was correct.
  - **Test case 2** : as a user, in the insert page, I enter the desired text for each column. Then I push the enter button.  
Correct Output : The app displays a toast message saying insert was correct.
  - **Test case 3** : as a user, in the delete page, I first select the data I want to delete. Then I push the delete button.  
Correct Output : The app displays a toast message saying the delete was correct.
- **Feature 2 Test Cases** : as a user, I want to Update, Insert, Delete rows relating to Burroughs.
  - **Test case 1** : as a user, in the update page, I input what row I want to edit. Then I enter the desired update in the update text box. Then I push the update  
Correct Output : The app displays the count of the number of records.
  - **Test case 2** : as a user, in the insert page, I enter the desired text for each column. Then I push the enter button.  
Correct Output : The app displays a toast message saying insert was correct.
  - **Test case 3** : as a user, in the delete page, I first input what I want to delete. Then I push the update button.  
Correct Output : The app displays a toast message saying the delete was correct.
- **Feature 3 Test Cases** : as a user, I want to Update, Insert, Delete rows relating to Time.
  - **Test case 1** : as a user, in the update page, I first input what I want to edit. Then I enter the desired update in the update text box. Then I push the update button.  
Correct Output : The app displays the count of the number of records.
  - **Test case 2** : as a user, in the insert page, I enter the desired text for each column. Then I push the enter button.  
Correct Output : The app displays a toast message saying insert was correct.
  - **Test case 3** : as a user, in the delete page, I first select the row I want to delete. Then I push the update button.  
Correct Output : The app displays a toast message saying the delete was correct.
- **Feature 4 Test Cases** : as a user, I want to Update, Insert, Delete rows relating to Vehicle Class.
  - **Test case 1** : as a user, in the update page, I input what row I want to edit. Then I enter the desired update in the update text box. Then I push the update button.  
Correct Output : The app displays the count of the number of records.
  - **Test case 2** : as a user, in the insert page, I enter the desired text for each column. Then I push the enter button.  
Correct Output : The app displays a toast message saying insert was correct.



- **Test case 3** :as a user, in the delete page, I first select the row I want to delete. Then I push the update button.

Correct Output : The app displays a toast message saying the delete was correct.

- **Feature 5 Test Cases** : as a user, I want to Update, Insert, Delete rows relating to Active Vehicles.

- **Test case 1** :as a user, in the update page, I input what row I want to edit. Then I enter the desired update in the update text box. Then I push the update button.

Correct Output : The app displays the count of the number of records.

- **Test case 2** : as a user, in the insert page, I enter the desired text for each column.Then I push the enter button.

Correct Output : The app displays a toast message saying insert was correct.

- **Test case 3** :as a user, in the delete page, I first select the row I want to delete. Then I push the update button.

Correct Output : The app displays a toast message saying the delete was correct.

- **Feature 6 Test Cases** : as a user, I want to Update, Insert, Delete rows relating to Street.

- **Test case 1** :as a user, in the update page, I input what row I want to edit. Then I enter the desired update in the update text box. Then I push the update button.

Correct Output : The app displays the count of the number of records.

- **Test case 2** : as a user, in the insert page, I enter the desired text for each column.Then I push the enter button.

Correct Output : The app displays a toast message saying insert was correct.

- **Test case 3** :as a user, in the delete page, I first select the row I want to delete. Then I push the update button.

Correct Output : The app displays a toast message saying the delete was correct.

- **Feature 7 Test Cases** : as a user, I want to backup file 1

- **Test case 1** : as a user, in the file 1 page, I select the backup to save the edited file

Correct Output : The app displays a toast message saying backup is completed.

- **Feature 8 Test Cases** : as a user, I want to backup file 2

- **Test case 1** : as a user, in the file 2 page, I select the backup to save the edited file

Correct Output : The app displays a toast message saying backup is completed.

- **Feature 9 Test Cases** : as a user, I want to backup file 3

- **Test case 1** : as a user, in the file 3 page, I select the backup to save the edited file

Correct Output : The app displays a toast message saying backup is completed.

- **Feature 10 Test Cases** : as a user, I want to backup file 4

- **Test case 1** : as a user, in the file 4 page, I select the backup to save the edited file

Correct Output : The app displays a toast message saying backup is completed.

- **Feature 11 Test Cases** : as a user, I want to import a edited version of file 1
  - **Test case 1** : as a user, in the file main page, I select the file I want to import  
Correct Output : The app displays a toast message saying import is completed.
- **Feature 12 Test Cases** : as a user, I want to import a edited version of file 2
  - **Test case 1** : as a user, in the file main page, I select the file I want to import  
Correct Output : The app displays a toast message saying import is completed.
- **Feature 11 Test Cases** : as a user, I want to import a edited version of file 3
  - **Test case 1** : as a user, in the file main page, I select the file I want to import  
Correct Output : The app displays a toast message saying import is completed.
- **Feature 12 Test Cases** : as a user, I want to import a edited version of file 4
  - **Test case 1** : as a user, in the file main page, I select the file I want to import  
Correct Output : The app displays a toast message saying import is completed.

**ToDo task list for the next sprint:**

Backlog	In progress	Testing	Done
Design GUI			
Backup Import for file 1			
Backup Import for file 2			
Backup Import for file 3			
Backup Import for file 4			
Update, Delete, Insert for column file 1			
Update, Delete, Insert for column file 2			
Update, Delete, Insert for column file 3			
Update, Delete, Insert for column file 4			

Test 1 - update file 1	Test 5 - Insert a row in file1	Test 6 - Delete a row in file 1	Test 6 - backup file 1	Test 7 - import file 1
select update	select update button	select delete button	Select back button	Select import button
A row should be changed	A row should be added	A row should be deleted	Changed file should be save	File 1 should be loaded in

**ToDo List:**

Done list of sprint 2

Backlog	In progress	Testing	Done	Who did it
			Design GUI - menu + search raw (Rohan, Jeanine)	Rohan + Sergio
			Rest API	everyone
			Servlets 1& 2	Sergio
			Servlets 3 & 5	Steven
			Servlet 4	Roz
			Servlet 6	Shikher